REPORT R-507 MAY, 1971 UILU-ENG 71-2210

CSL COORDINATED SCIENCE LABORATORY

FAULT DIAGNOSIS IN FET MODULES

M. PAIGE
G. METZE

UNIVERSITY OF ILLINOIS – URBANA, ILLINOIS

UILU-ENG 71-2210

FAULT DIAGNOSIS IN FET MODULES

by

M. Paige and G. Metze

This work was supported in part by the Joint Services Electronics Program (U. S. Army, U. S. Navy and U. S. Air Force) under Contract DAAB-07-67-C-0199, and in part by the International Business Machines Corporation.

FAULT DIAGNOSIS IN FET MODULES

M. Paige and G. Metze
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois

ABSTRACT

The use of Field Effect Transistor (FET) devices in logic design has changed the design emphasis from networks composed of single logic gates to networks composed of complex functional modules. Fault diagnosis techniques which have been discussed in the literature are based on the former type networks and hence are somewhat inadequate for this new technology.

This paper presents an approach to the generation of tests to detect all single and multiple faults of the stuck-line type in FET modules realizing complex functions. These networks are treated in a uniform manner, and a uniform notation is adopted for the tests required for diagnosis.

As the range of problems to which computing systems have been applied has widened, the task of ensuring that a computer is operating correctly has become steadily more difficult. The shift in networks from discrete component to integrated circuit realizations has caused widespread change in machine design. The shift from transistor to FET technology also presents a new change in design policy. The advent of FET technology has caused the usual gate-type design using AND, OR, NAND, or NOR gates to be replaced (or reclaimed) by the classical contact network design, since the latter approach is closer to the capabilities of this new technology.

Fault diagnostic techniques for FET networks therefore should reflect this type of logical capability. Compared to the extensive literature on fault diagnosis for gate-type networks (1), little work has been published for FET technology (8).

This paper presents an approach to the generation of fault detection tests for FET networks with certain design configurations. These networks can be treated in a uniform manner, and a uniform notation for the tests required can be adopted and used.

The diagram in Figure 1 is a simple bidirectional FET gate of the kind that will be considered here. When the control lead is sensitized, current flows in the direction forced by the network environment. Hence these devices are bidirectional and their behavior is identical to that of contact networks treated in Shannon's 1938 paper. Thus, relay network analysis and synthesis techniques of the early literature can now be used again. The operation of these devices

suggest an obvious fault model which can be used for analysis. The failure mode for permanent faults will be the device stuck-at-short (sa1) or stuck-at-open (sa0). This model has been substantiated by engineering analysis (7).

FET technology can be used to realize conventional logic gates. However, because of the high packing density available they can be better used to realize more complex Boolean functions. One such complex configuration which can be readily implemented using FET gates is the "bridge" network as shown in Figure 4. The technique presented here for deriving fault detection tests for complex FET networks is a natural extension of a technique developed for conventional logic networks (4,5).

The model in Figure 2 is assumed for a conventional FET network module. We will consider the module as containing two parts: a contact-like network producing a function $F(x) = F(x_1, x_2, \ldots, x_k)$, and an inversion scheme which yields the output function $\overline{F(x)}$. The inversion is provided by a drive which is used for both loading and to make these modules functionally complete, that is, to make it possible to realize all Boolean functions with them. The network in Figure 3 is typical of FET circuits. It will be assumed that, if a given variable is used within the module in both its true and complemented form, each form will use a separate input line. This restriction is not unreasonable since any complementation of a given variable must come from outside the module (from a previous module or double rail logic).

The following definition will be used to characterize the networks considered in this paper.

Definition 1. A network is <u>irredundant</u> if the output function of the network expanded into sum-of-products form, e.g.,

$$F(x) = P_1(x) + P_2(x) + \ldots + P_n(x)$$

where $P_i(x)$ is a product term, has the following properties:

(a) there is a $P_i(x)$ for every path through the network;

(b) $P_i(x) \neq P_j(x)$ for any i,j except i = j;

(c) no literal or combination of literals can be removed from any $P_i(x)$ without changing the function.

According to this definition, F(x) describes an equivalent network consisting of a parallel connection of all paths through the original network. For example, $x_2\bar{x}_3 + x_1\bar{x}_2 + \bar{x}_1x_3$ and $\bar{x}_1x_3 + \bar{x}_1x_2 + x_1\bar{x}_2$ are both irredundant forms of the same function. Note that the structure of the network is reflected in the output expressions, that is, these two expressions represent two different network configurations, both of which are irredundant.

We consider only networks which are irredundant by the above definition. To determine if a given network is irredundant, the output expression of the network, F(x), is expanded into sum-of-products form. If this result cannot be simplified by cancellation or absorption, then the network is irredundant. The restriction to this type of network should not be startling since it has long been recognized that redundancy only hinders fault detection.

Definition 1 can be interpreted along the following line: if a network is irredundant then no element in the network can be removed

or replaced by a constant value without changing the output function. Hence any fault in the network must cause some net change in the output function.

The following definition will help to simplify the testing procedure.

Definition 2. A function F(x) written in sum-of-products form is said to be positive if it contains only uncomplemented variables.

It is obvious that the contact-like portion of an FET module can only realize a positive function F(x) of the input variables $x_1, x_2, \ldots, x_k$ to the network. Therefore, any complemented variables originally in F(x) must be supplied externally and must be used as new input variables. For example, in Figure 3 the output function F(x) could be written as

$$F(x) = P_1(x) + P_2(x) + P_3(x),$$

where $P_1(x) = x_2x_6$, $P_2(x) = x_1x_5$, and $P_3(x) = x_3x_4$, with $x_4 = \bar{x}_1$, $x_5 = \bar{x}_2$, and $x_6 = \bar{x}_3$.

A binary vector representation for each product term, $P_i(x)$, of F(x) will be used to simplify the test generation procedure. We assume that there are k input lines to the network as shown in Figure 2, and thus all vectors will be of length k. For a given $P_i(x)$, a characteristic vector C_i is formed, where $C_i = (c_{i1}, c_{i2}, \ldots, c_{ik})$ and $c_{ij} = 1$ if and only if x_j is a factor in $P_i(x)$. For example, the output function in Figure 3 would have the following C-vectors:

$$C_1 = (010001) \qquad C_2 = (100010) \qquad C_3 = (001100).$$

Note that the commas have been omitted from the individual vectors.

Forming the C-vectors as mentioned above, it is not difficult to see that for a given F(x), $F(C_i) = 1$. Hence these vectors are true vectors of the function F(x).

We also form a set of vectors from these C-vectors, called single-change vectors, denoted by S_j. The S-vectors are formed by systematically replacing each 1 in the C-vector by a 0. Thus, each C-vector generates as many S-vectors as the number of 1's it contains. For example, C_1 = (010001) generates two S-vectors, (000001) and (010000). For convenience these S-vectors will be subscripted from 1 to m, where m is the total number of 1's in the C-vectors.

We have already seen that the C-vectors are true vectors for F(x); the following theorem establishes the use of the S-vectors.

Theorem 1. Given that $F(x) = P_1(x) + P_2(x) + \ldots + P_n(x)$ is a positive irredundant function, then no S-vector is a true vector of F(x). ($F(S_i) = 0$ for all i.)

Proof:

Since F(x) is a positive function, then a given S-vector, if it were a true vector, would be a true vector of a positive product term. This product term because of the construction of the S-vectors from the C-vectors (hence from a product term already in F(x)) would necessarily include one of the original product terms. Hence an S-vector can not be a true vector of any product term in F(x), since this would contradict the irredundancy of F(x). Therefore no S-vector is a true vector of F(x). QED

It has been recognized that if a network is irredundant (2,6) that there exists a single fualt test set which detects all multiple faults. Using the FET fault model described earlier, it is obvious that each network fault is represented by a set of input conditions to various network elements. Hence each fault corresponds to some logical change in the output function.

Each single fault in an irredundant network corresponds to a change in one or more product terms of the output function. We can characterize these changes in each product terms as one of two types as shown by Paige (4,5):

(a) the product term becomes a logical 0;

(b) the product term becomes independent of a single input variable, that is, that variable is not represented in the new product term.

If condition (a) exists then the C-vector for that product term represents a test for it, since this vector gives the output 1 if and only if that product term is present. If condition (b) exists then a new product term is formed from a given term, that is, a new product term is formed which represents the original term independent of a given variable. The S-vectors are the true vectors of all such possible new product terms, hence the S-vectors will not give the output 0 for F(x) if one of these new product terms exists. Thus the S-vectors represent tests for condition (b).

The test set made up of C- and S-vectors will detect all multiple faults in an irredundant network. This result has been shown

by Paige (5) and is based on the notion that in an irredundant network a multiple fault can be detected when its single contributing faults are. This result has been shown in the reference cited to depend heavily on the single fault test set selected. The C- and S-vectors meet the criteria which have been determined for that test set.

The network in Figure 4 is a bridge circuit; a configuration which, as previously mentioned, is uncommon in logic-gate networks but can be used in FET networks. The output function of this network,

$$F(x) = x_1x_4 + x_1x_3x_5 + x_2x_5 + x_2x_3x_4,$$

is irredundant. Hence the following set of tests will detect all faults:

C-vectors:	(10010)	(10101)	(01001)	(01110)
S-vectors:	(00010)	(00101)	(00001)	(00110)
	(10000)	(10001)	(01000)	(01010)
		(10100)		(01100).

These 14 tests detect all single and multiple faults in the network. This constitutes a reduction of 18 over the total number of input combinations. Further reduction can be made on the number of S-vectors, since some combinations are allowed; however, these procedures are bookkeeping only (5).

We have presented a test generation procedure for FET modules in this paper. The method is easy to employ and is efficient; based on an irredundancy criterion the tests detect all single and multiple faults.

If a logic network is constructed which proves to have undetectable faults, the failures can compound to eventually cost more

in time and effort than a redesign would require. It appears evident (to some of us) that diagnosis should be considered as a design issue, that is, some conscious effort should be made to incorporate reasonable diagnostic properties into the specification of any hardware. The work presented here is part of a continuing study of the relationship between logic design and fault diagnosis. Further work is being done with two-level FET networks as recently proposed by Ibaraki and Muroga (3).

References

1. Chang, H., Manning, E. and Metze, G., Fault Diagnosis of Digital Systems, Wiley-Interscience, 1970.

2. Hayes, J., "A Study of Digital Network Structure and Its Relation to Fault Diagnosis," University of Illinois, Coordinated Science Laboratory Report R-467, May 1970 (CFSTI AD 707 691).

3. Ibaraki, T. and Muroga, S., "Synthesis of Networks with a Minimum Number of Negative Gates," IEEE Computers, January 1971, pp. 49-58.

4. Paige, M., "Generation of Diagnostic Tests Using Prime Implicants," University of Illinois, Coordinated Science Laboratory Report R-414, May 1969, (CFSTI AD 688 832).

5. Paige, M., Doctoral Thesis (in progress).

6. Schertz, D., "On the Representation of Digital Faults," University of Illinois, Coordinated Science Laboratory Report R-418, May 1969 (CFSTI AD 688 836).

7. Schnable, G. and Keen, R., "Failure Mechanisms in Large Scale Integrated Circuits," IEEE Electron Devices, April 1969, pp. 322-332.

8. Yen, Y., "Computer Aided Test Generation for 4 ϕ MOS LSI Circuits," IEEE Computers, October 1969, pp. 890-893.

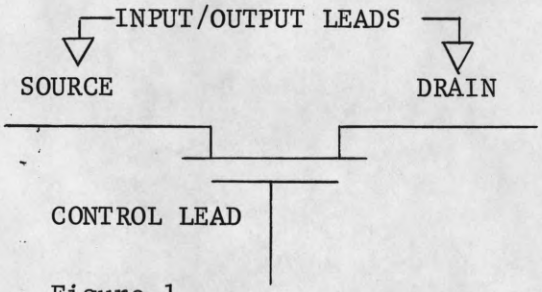

Figure 1.
Bidirectional FET Gate

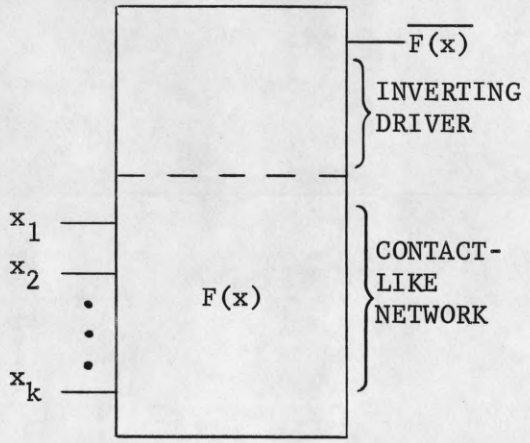

Figure 2. (right)
FET Network Model

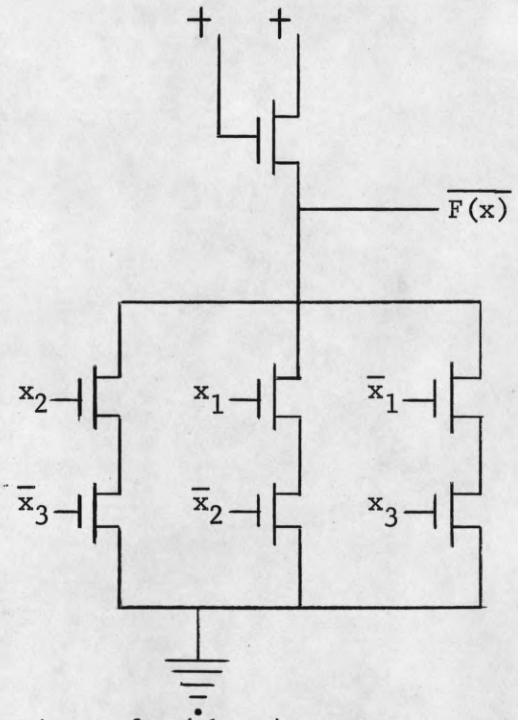

Figure 3. (above)
FET Network
$F(x) = x_2\bar{x}_3 + x_1\bar{x}_2 + \bar{x}_1x_3$

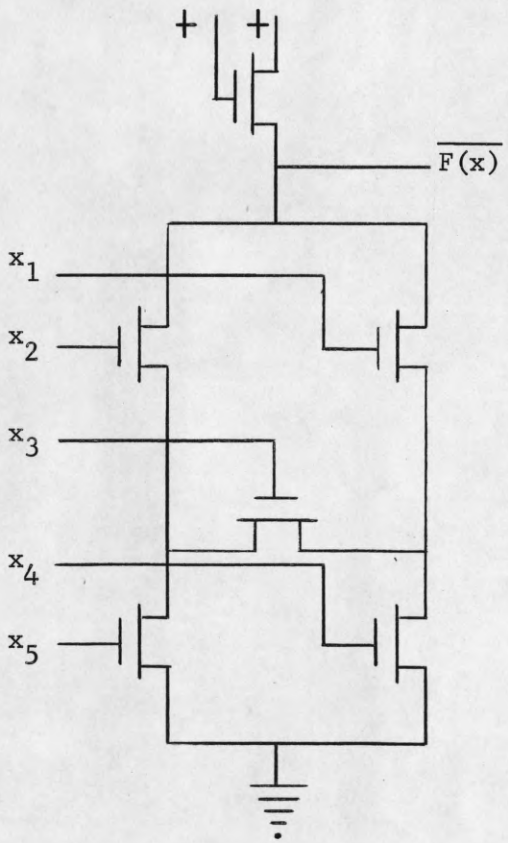

Figure 4. (right)
FET Bridge Network
$F(x) = x_1(x_4 + x_3x_5) + x_2(x_5 + x_3x_4)$

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP

3. REPORT TITLE
FAULT DIAGNOSIS IN FET MODULES

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*
M. Paige and G. Metze

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May, 1971	9	8

8a. CONTRACT OR GRANT NO.
DAAB-07-67-C-0199; International
b. PROJECT NO. Business Machines Corporation
c.
d.

9a. ORIGINATOR'S REPORT NUMBER(S)
R-507

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
UILU-ENG 71-2210

10. DISTRIBUTION STATEMENT
This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Joint Services Electronics Program through U. S. Army Electronics Command, Fort Monmouth, New Jersey 07703

13. ABSTRACT

The use of Field Effect Transistor (FET) devices in logic design has changed the design emphasis from networks composed of single logic gates to networks composed of complex functional modules. Fault diagnosis techniques which have been discussed in the literature are based on the former type networks and hence are somewhat inadequate for this new technology. This paper presents an approach to the generation of tests to detect all single and multiple faults of the stuck-line type in FET modules realizing complex functions. These networks are treated in a uniform manner, and a uniform notation is adopted for the tests required for diagnosis.

DD FORM 1 NOV 65 1473

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Fault Diagnosis						
Fault Testing in Combinational Logic						
Multiple Faults						